The Case for a Portuguese Web Search Engine

Mário J. Silva

DI-FCUL TR–03–3

March 2003

Departamento de Informática
Faculdade de Ciências da Universidade de Lisboa
Campo Grande, 1749-016 Lisboa
Portugal

The Case for a Portuguese Web Search Engine

Mário J. Silva, mjs@di.fc.ul.pt

Universidade de Lisboa, Faculdade de Ciências

Campo Grande, 1749-016 Lisboa, Portugal

March 14, 2003

Abstract: This paper presents the case for a search engine for the Portuguese Community. Preservation of publications of historical interest for future access, obtaining knowledge about the preferences and interests of our society in the information age and intelligence gathering for security and protection are examples of national interests addressed by such system. This paper also introduces the architecture and design of *tumba!,* a new Internet search engine for the Portuguese Web. Tumba! proposes a new repository architecture and uses innovative ranking and presentation algorithms. It is designed to take profit of our unique knowledge of this web making it a platform that could work as testbed for research and evaluation of new ideas in Web information retrieval and the computational processing of Portuguese.

1 Introduction

The Web, as laid on the public Internet, has a global nature. However, the behaviour of Web users reflects their social background and their linking patterns reflect how they group in communities (see Gibson et al., 1998). Web communities can be geographically spread across the globe, its members may have many interests and belong to several unrelated communities, but there is a distinct language, vocabulary or other social behaviour characteristic that binds them through the web linkage. This has been identified through analysis of connectivity patterns on the web global graph (see, for instance, Kumar et al., 1999).

The "Portuguese Web" is the subset of the pages of the Web that are related to Portugal. It is not easy to give a precise definition of what constitutes the Portuguese Web, but intuitively everyone understands this concept as the collection of information of relevance to the Portuguese people on the Internet. Throughout this paper, I will discuss the Portuguese Web having in mind its representation as the collection of pages from the global web that satisfy one the following conditions:

- Hosted on a site under a ".PT" domain.

- Hosted on a site under other domain (except ".BR"), written in Portuguese and with at least one incoming link originating in a web page hosted under a ".PT" domain.

Tumba! (Temos Um Motor de Busca Alternativo) is a Web search engine specially crafted to provide better results to those searching information on the Portuguese Web. It incorporates knowledge about the usage profile and interests of those who access these pages to improve Web searches. Tumba! is being offered as a public service since November 2002 (see http://www.tumba.pt).

I believe that the implementation of a search engine specialised in providing services to community webs, such as tumba!, is a valuable alternative to global search engines for locating information within the Web pages of these communities. The lack of context or scope information in queries submitted to global engines causes that users in different communities will perform the same query and expect completely different results when a search is made, in particular when these tend to consist of a small number of words which mean different things in different places/languages (see, for instance, the results given by global search engines queries on "passaporte" or "biblioteca"). Tumba! has a similar architecture and adopts many of the algorithms of global search engines, but its configuration data is much richer in its domain of specialisation. It has a better knowledge of the location and organization of Portuguese web sites (both in qualitative and quantitative terms).

As no other Portuguese organization is systematically crawling and archiving the contents of this Web, the importance of the development of tumba! has more than a simple cultural or commercial interest: it may become strategic both for government entities and national industries as the resource for locating information and services for web users communicating primarily in the Portuguese language or interested in locating resources related to Portugal.

Tumba! is now the result of about two years of development by the members of the XLDB research group of LASIGE – Large-Scale Information Systems Laboratory of Faculdade de Ciências da Universidade de Lisboa[1]. This paper discusses the motivation

[1] see http://xldb.fc.ul.pt

for building and offering tumba! as a public service, the requirements that have driven its architectural design and presents the case for search engines directed to community webs.

The paper is organized as follows: first, I will quickly review the less that 10 years long history of Web search engines. Section 3 details the architecture of tumba! and Section 4 summarizes what has been accomplished. In Section 5, I discuss the motivations for operating a Portuguese Web search engine as one instance of a community search engine. Finally, I will discuss some of the directions for future improvements to tumba!.

2 Web Search and Archiving

Web search engines are almost as old as the Internet (see Arasu et al., 2001, and Kobayashi and Takeda, 2000, for detailed reviews). The first search engines applied pre-web information retrieval concepts. The web was seen as a collection of documents, each abstracted as a simple bag of words extracted from the HTML source. Searches were performed against an index of the words built from this collection. The search model was based on the Vector Space Model for Information Retrieval, which computes the importance of a document to a given query as the level of similarity between the query and the document, where the similarity is the dot product of the vectors formed taking each term as an independent dimension in a R^n space, weighted by a measure of the relative importance of the search terms within each document (see Salton, 1989).

The quality metric of information retrieval systems, designated as relevance, is measured on the importance to the users' needs of the documents returned by their queries. As the web grew, finding relevant pages using the Vector Space Model became harder. Soon search engines started tuning their algorithms to the specificities of the web data: words in the title of documents, close to the top or in some special HTML meta-tags received higher weight than other words within HTML pages. Examples of such engines include Altavista[2] and NorthernLight[3].

Web search technology eventually evolved to use a new search paradigm. State-of-the-art search engines now rank web pages based on concepts taken from bibliometrics that

[2] see http://altavista.com

[3] see http://norternlight.com

provide much better results, such as the measure of documents authority. This is computed as the number of pages that link to a given page from the set of pages that match a query (see Kleinberg, 1998). The most widely used algorithm of this kind is PageRank (see Page, 1998). Unlike the authority measure computed with Kleinberg's algorithm, PageRank computes a measure of the popularity of every page based on the number of pages referencing it. As this measure is computed offline, it can be used in fast ranking computations, making it suitable for global search engines. PageRank builds a graph of the web, where pages are nodes and edges the links that connect those pages, and then computes the importance of every page as the number of links that recursively point to a page weighted by the importance of each referencing page. The PageRank algorithm computes authority values iteratively. It has been shown that these converge to the eigen values of the connectivity matrix for the web graph. PageRank is used in Google[4], the most widely used web engine today.

Another dimension of Web search engines lies in the capacity to preserve the successively crawled pages for each web site in an historic database. The largest effort to archive the Web is an initiative of the Internet Archive (see Kahle, 1997). There is currently no service providing both an interactive search interface and an archival service at the global level. Internet Archive stores the sites it crawls over time but does not offer a search facility or a way for navigating in the pages collected during a certain period of time. There are also several initiatives of national libraries to archive portions of their webs (See Day, 2003, for a recent report covering this topic).

After a period of growth, the number of global search engines is decreasing. Only 6 global search engines remain: Alltheweb, Altavista, Google, Inktomi, Teoma and WiseNut. Recently, Yahoo acquired Inktomi and Overture acquired both Altavista and Alltheweb, showing that independent search services companies are becoming part of or turning into advertising networks. Global web search facilities are now in the hands of a few private companies, all based in the United States.

The market for search engines is by no means stable. Since the beginning of the explosion of the Internet we have been witnessing the arrival of new players that introduce technical innovations that set new standards for information search. The

[4] see http://google.com

advent of the Semantic Web (Berners-Lee et al., 1992) and an increasing trend for publishing useful, credible and accurate meta-data by public sector organizations could be the trigger for a new generation of search technologies incomparably more useful that what we have today.

Web search tools aren't limited to engines that crawl the web, such as the ones presented. In addition to these, there is a multitude of engines that provide directory services or tend to combine directory listings with web search results. There is innovative research in topic distillation, ie, the retrieval of collections of relevant documents for a given information need (not necessarily containing search terms) (see Bharat and Henzinger, 1998). On the other hand, Vivissimo[5] is one example of a meta-search engine that is able to cluster search results interactively, based on the information provided in the snippets of the results of other search engines.

3 Tumba!

Tumba! implements some the best know algorithms for web search and is pushing the envelope in some domains. Tumba!'s repository is built on a novel framework for parallel and incremental harvesting of web data, based on version management (see Katz,1990). In addition, it can provide better rankings than global search engines, as it makes use of context information based on our local knowledge of the Portuguese web and our handling of the Portuguese language. A test version of tumba! is now also offering a beta version of a new interface that will provide an hierarchical organization of search results using a clustering algorithm (see Martins, 2003).

A substantial part of the tumba! software was initially developed for other projects. A selective harvesting system for the digital deposit of Web publications for the National Library of Portugal (see Noronha, 2001) has provided the environment for building the first prototype of tumba!. In parallel, the development of XMLBase[6], a Web data warehousing system began to take shape. XMLBase is a component-based web data management system, including a meta-data repository, a content server and associated ETL (Extract/Transform/Load) software for HTML and XML data management. This

[5] http://vivisimo.com

[6] see http://xldb.fc.ul.pt/xmlbase

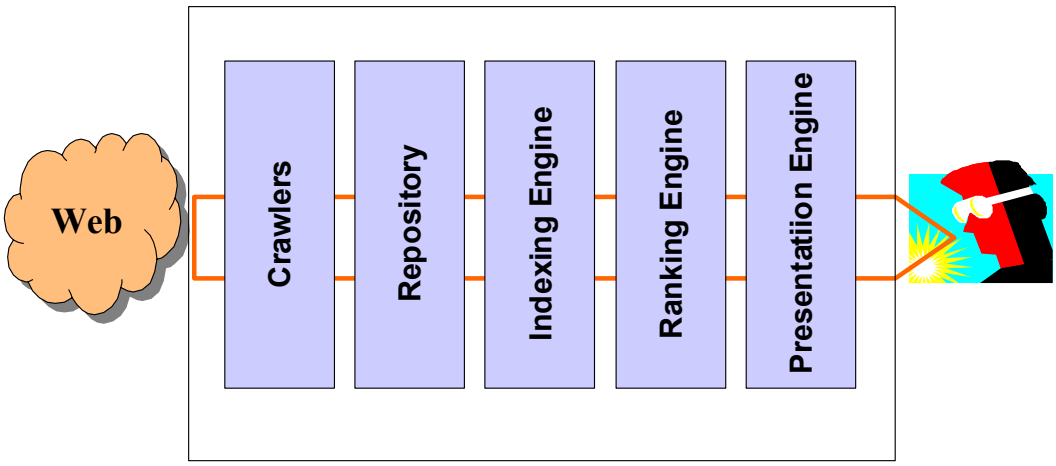

Figure 1 - The main components of tumba! create a pipeline that transforms information collected from the web in successive stages.

had strong implications and influenced the software architecture of tumba!. The past experience has shown that many of the blocks that compose the current system could be reused and combined to form the basis of a wide range of Internet applications. Tumba!'s technology could be useful for software applications demanding:

- selective harvesting of web sites;
- complex, high-performance, web data warehouse systems;
- error-tolerant parsing of badly formed HTML;
- new, XML-aware data storage, indexing and search systems.

The concept of a web crawler as a reusable software component was first proposed in Mercator, the crawler used by the Altavista search engine (see Najork and Heydon, 2001). The Google "anatomy" paper also provided a good source of inspiration for defining the functionality of some of the main software blocks of tumba!. However, the APIs of some of tumba!'s components are radically different, in particular the organization of the repository and crawling software.

The remainder of this section details the software architecture of tumba! and its main components.

3.1 Architecture

The architecture of tumba! follows the pipelined (tiered) model of high performance information systems. One main difference between tumba! and other search engines is on the emphasis on reusable software components. Some of the main data management

software components that have been incorporated in tumba! are commercial products, while others are available as open source. Some of the components of tumba! were conceived to be used independently by other web data processing applications.

Tumba!'s data pipeline is illustrated in Figure 1. Information flows from web publishing sources to end users though successive stages. At each stage a different transformation is performed on the web data:

- In the first stage, ***crawlers*** harvest web data referenced from an initial set of domain names and/or web site addresses, extract references to new URLs contained in that data, and hand the contents of each located URL to the web repository.
- The Web ***repository*** is a specialised database management system that provides mechanisms for maximizing concurrency in parallel data processing applications, such as Web crawling and indexing. It can partition the tasks required to crawl a web space into quasi-independent working units and then resolve conflicts if two URLs are retrieved concurrently. This mechanism is implemented upon a versioning model, derived from the conceptual models of engineering databases.
- The ***indexing engine*** is a set of services that return a list of references to pages in the repository, matching a given list of user supplied keywords.
- The ***ranking engine*** sorts the list of references produced by the indexing engine by the perceived relevance of the list of pages.
- The ***presentation engine*** receives search results in a device-independent format and formats them to suit multiple output alternatives, including web browsers, mobile phones, PDAs and Web Services. The presentation engine can also cluster search results in multiple ways.

I will now describe each of these main components in more detail.

3.2 Web Crawler and Repository

The web crawling and data repository sub-system of tumba! has three main components:Versus, a Web meta-data repository; VCR, the Versus Content Repository; and Viúva Negra, a Web crawler built upon Versus.

Versus and VCR are components of XMLBase, a semi-structured research data management system under development at the XLDB group. Versus provides a framework for storing large quantities of information retrieved from the web throughout time and give access to it to a range of data and knowledge management tools. The objectives of this framework support many of the needs of tumba! (see Campos et al., 2002 and Gomes, D. et al., 2002):

- harvesting large quantities of data from the Web;
- managing meta-data about large collections of web resources;
- organizing Web data in a temporal dimension.

Versus performs many Web data management tasks and provides support for parallel execution of many operations, through a versions and workspaces model, data partitioning functions and time support. Versus is designed to maintain information about Web objects. In Versus, a Web object is an entity that has an associated URL and may contain references to other Web objects. The main classes of the Versus data model are:

- ***Workspaces*** represent collections of objects. These are shown with 3 different levels of visibility to applications. Private workspaces are exclusive of individual application threads; Group workspaces are shared by application threads and reconcile data received from Private workspaces; finally, Archive workspaces are append-only and can be read by any Versus application. Data can be copied/moved between workspaces through check-out and check-in operations.
- ***Versions*** capture each one of the representations (or "images" – the contents of URLs) that a Web object may have over time. Versions may have multiple ***Facets***, which store alternative views of objects: a given Web object may have a facet containing its original publishing format (in HTML or PDF), a text representation consisting of the extracted words, or a well-formed XHTML document generated from the captured data.
- ***Layers*** represent collections of Versions, with the restriction that they may contain only one version of each object. Versions may have an associated label

or time stamp, representing collections such as “the web pages crawled in March 2003.”

- ***Partitions*** represent schemes for dividing the objects represented in disjoint groups called Working Units. An example partition would be “the set of hosts on the web”, where each Working Unit would then represent the set of all Objects with URLs that share the same host component.

Applications operate under the following model: first, they define a Partition on a Group Workspace and set a current Layer. Then, they start a group of application threads, where each thread processes a Working Unit at a time in its own Private Workspace and checks the result back in the Group Workspace, associating it with the current Layer. When all Working Units have been processed, the Application may end or instruct Versus to check-in the data in the current layer of the Group Workspace into an Archive Workspace.

When an application needs to store the contents of an URL, it uses Versus services to process the download. Versus is designed to operate above a data store organized as a distributed file server that stores the contents of retrieved objects and any additional objects that may be derived from these during the post-processing tasks that follow. In tumba!, the VCR – Versus Contents Repository, provides that service. The VCR is a distributed storage server for the Web objects loaded and managed by Versus, which hides the physical details of data location and replica management of the archived contents.

The crawler used by tumba!, ViúvaNegra, is derived from a previously developed crawler, Tarântula, which had its own database (see Gomes and Silva, 2001). This new crawler makes extensive use of many of the features provided by Versus, including support for managing the parallel processing of web data and management of its history.

When operating with Versus, Viúva Negra detects contents that are replicated in the repository (by computing and maintaining MD5 hashes for each stored contents). This makes it suitable to process successive crawls of web spaces without the need to maintain complete replicas of all the crawls. At the same time, we keep in our repository full copies of all the crawls.

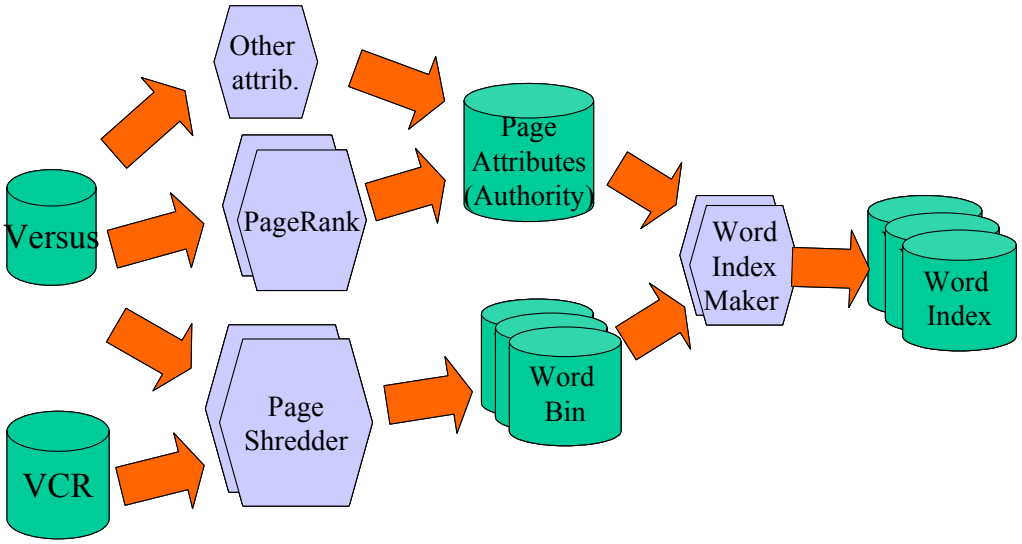

Figure 2 - The off-line process of Sidra involves parallel shredding of the pages to index, and parallel generation of word indexes. Document references in word indexes are sorted in popularity order.

3.3 Indexing and Ranking

Indexing and Ranking in tumba! are performed by SIDRA, a new indexing and ranking system (see Costa, 2003). The initial version of the indexing engine of SIDRA is based in Oracle Intermedia[7], with ranking computation using custom algorithms (see Costa and Silva, 2001). This version is still in operation and the public website of tumba!

However, as Intermedia was designed to index corporate webs, it is hard to adapt it to index large-scale networks as required by tumba!. The inverted file within Intermedia sorts the document references associated to each word by criteria that do not match the final ranking order that tumba! outputs, which is largely weighted by the relative importance of the pages as inferred from the analysis of the web linkage. This implies that on-line sort operations in ranking computations are much more complex than they would if document lists were pre-sorted by tumba!'s measure of popularity.

In addition, Intermedia does not enable the addition of tumba!-specific attributes to the indexed terms, which will be essential in the next planned versions of the tumba! software.

---

[7] see http://www.oracle.com/intermedia.

The design of SIDRA reflects the requirements derived from the experience of building and operating tumba! with a proprietary index. Like in any other index, SIDRA runs both off-line and on-line. Most of the expensive computing tasks are performed off-line. These include (see Figure 2):

- retrieving the list of documents and some of their meta data from Versus, and the documents' contents from VCR;
- *shredding* documents into word bins. A program, called the PageShredder, scans the page references associated with a Layer in a Versus workspace to initiate the term index generation process. It uses the data partitioning capabilities available and processes each unit by an independent thread under the control of Versus. In the end of the shredding process, each bin is assigned to a specific word-range and will contain the set of document references to associate to each word;
- computing the importance of each page, using our implementation of the PageRank algorithm. Versus provides the linkage data to create the graph of the web on which the static ranking (or importance) of each page is computed. In the end, the results are stored in a database that contains the information attributes associated to each page that will be necessary to compute the final rank;

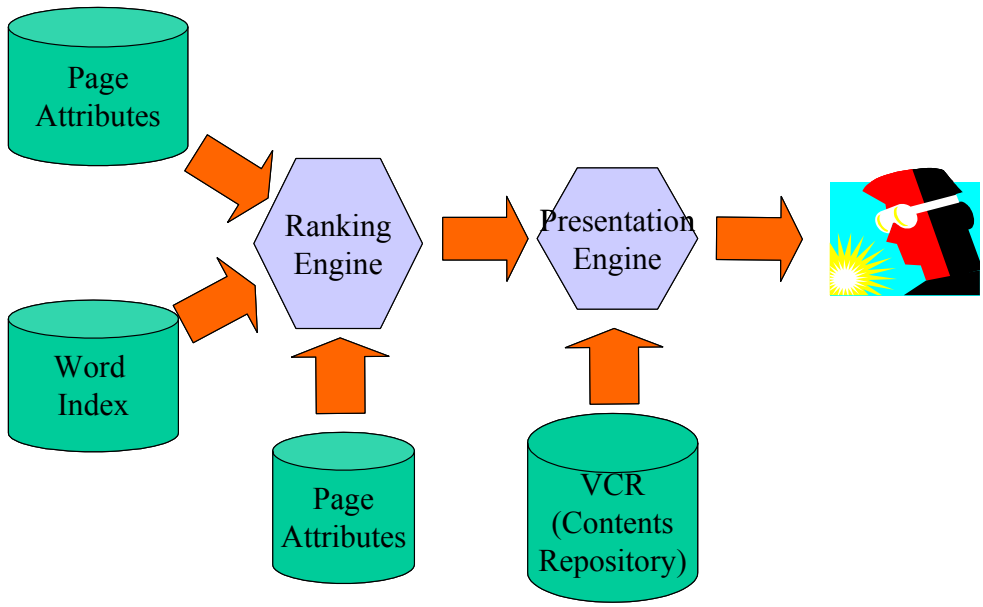

Figure 3 - Architecture of the on-line subsystem of tumba!

- creating word indexes with WordIndexMaker. This program takes a word bin as input and creates a corresponding word index as a compressed term-document structure. In a word index, each document receives an identifier that reflects its static rank. Each word entry in the word index has the documents sorted by the value of these identifiers.

The online part of SIDRA takes the word indexes as input and computes the final ranking of results to present in response to a user's query. These results are then pipelined to tumba!'s Presentation Engine (See Figure 3).

The information retrieval model adopted by tumba! combines the Vector Space Model with the Boolean Model (see Baeza-Yates and Ribeiro-Neto, 1999). The Ranking Engine starts by obtaining the pages with the highest static ranking page matching a query formulated as a Boolean expression of terms. This operation is designed to run with high parallelism. Word indexes are accessed through by QueryServers. Each QueryServer is configured to provide the page identifiers contained in one or more word indexes. There is no boundary on the number of Query Servers: they can be freely added to match the load and fault-tolerance levels intended for the system.

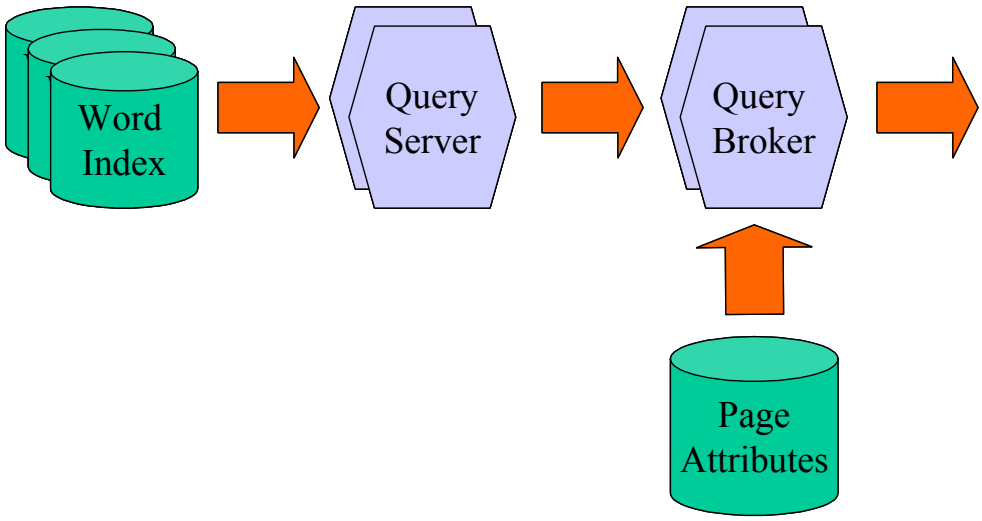

Figure 4 – Architecture of the Ranking Engine of SIDRA!

As QueryServers only have information about a range of terms, queries involving multiple terms may require merging the output of different servers. This is the task of QueryBrokers. QueryBrokers select the QueryServers that will be requested to provide results (based on the input query terms and load-balancing criteria) and combine the results by applying the logic restrictions specified in the input. As the page references in word indexes are pre-sorted, this can be performed very efficiently with a sort-merge-join algorithm.

At this stage, the tumba! ranking reflects the static page ranking (computed off-line). The experience with the evaluation of initial prototypes of tumba! has shown that the set of the top 200 results of the static ranking contains the set of the top 10 results of the optimal ranking (to be presented at the output). The final stage consists in obtaining these 200 results and changing their sort order to reflect dynamic scores that weight factors such as the presence of query terms in the title or description meta-tags of web pages (see Figure 4).

3.4 Presentation

The final stage of tumba! involves the generation of the page snippets that show the context of the use of the matching input query terms on the web pages. This is performed by obtaining from the Versus Content Repository the text previously extracted from each page/document and “greping” the relevant parts.

By default, tumba! presents results following the style that has been adopted since the first web search engines were created: output is in ranking order, through a filter that excludes from the listing those pages whose host is the same as in the previous result. A test version of tumba! is already demonstrating an alternative presentation interface using a page clustering algorithm tuned for the Portuguese language (see Martins, 2003).

Tumba! provides several user interfaces, supporting different output formats, including HTML and WML (for mobile phones). There is also a simplified HTML version for PDAs available[8].

4 Implementation, Operation and Usage Details

The tumba! search engine has evolved into a complex software system. Presently, it provides most of the features that are available on global search engines, including related pages search (see Dean and Henzinger, 1999), clustering of search results, a query spell-checker, links to Portuguese language dictionaries and a sub-system that enables users to listen to the pronunciation of query terms (developed by the Spoken Language Systems Lab of INESC[9]).

The software of tumba! can crawl and index not only HTML files, but also the most popular data types containing text information on the Web: Adobe PDF and PostScript, Microsoft Office and Macromedia Flash.

The existing infrastructure is presently composed of 3 medium-scale servers plus 6 older servers that are used for crawling. All the machines run the Linux operating system. A substantial part of tumba! is built on open source software:

- Versus is a software library that implements a Java API that is invoked by its applications. The internal data model for maintaining the metadata is relational, supported in Oracle9 and Hsql database servers.
- The Indexing software is written in C++ and Java and uses BerkeleyDB databases.

---

[8] See http://movel.tumba.pt

[9] See http://www.l2f.inesc.pt

- The Ranking and Presentation engines are written in Java and use the XML manipulation features of the Apache Cocoon publishing framework.

With no investment in advertising and very little visibility given by the media, tumba! is now serving several thousand queries/day and is supporting restricted searches from about 50 web sites that have added forms for searching on tumba!

It is hard to compare tumba! against other search engines, because there is either no information about their details of operation or the data that they have indexed. An analysis of the coverage (i.e., the number of documents indexed by Google, AlltheWeb and tumba!) is an apples to oranges comparison: the types of documents indexed differ and there is no data describing how they are distributed, crawls have been obtained at different times, and empirical evidence shows that the number of dead links referenced by these engines are substantially different.

The data in the tumba! repository is available for use by other researchers. One of the Versus applications is a dump utility, that can output a stream with the data stored in the repository in XML format. It provides many filtering options for selecting text, links, HTML meta-data attributes, etc. It has been used to provide input to the software used in another project to build a linguistic corpus of the Portuguese Web (See Fuchs, 2003).

The Portuguese computational linguistics community is organizing an initiative in evaluation of information retrieval systems processing information in Portuguese (See Aires, 2003). Tumba! will be evaluated and is offering part of its data, in the form of queries and Web document collections for other evaluation tasks.

The tumba! access statistics are also available (on demand) to researchers interested in studying the interest profiles of Portuguese users. This data shows that, while sex and related keywords dominate, as in global search engines, the most frequently used keywords reflect the pulse of the Portuguese society: they relate to events in business, environmental issues and national celebrities. On the other hand, the access log data also shows that the user population reflects the origins of tumba!, which is to a large extent composed of university students and faculty that began to use tumba! as a search tool in their daily activities.

Accesses to tumba! come from all over the world: in the first two weeks of March 2003 we accounted search requests from 74 top-level domains. However, the largest number of accesses are originated from Portugal: 78% of the accesses from resolved IP

addresses come from Portuguese service providers (this percentage computed from 65% of the total IP addresses accounted for).

5 The Importance of a Portuguese Community Search Engine

As discussed above, state of the art global search engines compute relevance using both word significance and a measure of the documents' importance inferred from the links'structure. However, these make little use of additional knowledge that can be extracted or inferred from the web. The developers of tumba! can identify the most important web sites on the Portuguese Web and obtain additional information for each of these that can be useful to improve the quality of search results. We could, for instance, easily characterize the most popular sites of the Portuguese web in terms of contents, the geographic scope of pages related to specific locations and embed this knowledge in our ranking algorithms.

We could also provide access to the deep and hidden webs of the most important web sites of our community, providing references to enter into content-rich databases in results pages. Recently, in a joint project with the national library of Portugal, we added to the test version of tumba! a demonstration that presents, when the name of a book author is detected on a query string, a result box with a link to access directly the records about that author in Porbase (the national on-line catalog of the Portuguese public libraries[10]).

The quality of search results can also benefit from the application of Portuguese linguistic analysis tools to our contents. Application of named entity recognition tuned for the Portuguese context could be useful in providing more quality results to the end-user. Geographic names recognition could in particular be useful to help mobile users performing location-related queries. This could be specially helpful for mobile users performing searches from small appliances (see Silva and Afonso, 1999).

Ranking computation is based on the assumption that web pages authors write them without a concern for how they will be ranked by search engines. However, as search engines became the most widely used web application, some web pages authors deliberately attempt to deceive search engines using diverse techniques (see Henzinger

[10] http://porbase.bn.pt

et al., 2002). Our deeper knowledge of the Portuguese web will enable us to better identify those sites that attempt to distort ranking results.

Search engines are becoming so effective in retrieving the names of sites that registration of appealing or easy to remember domain names is no longer important. The replacement of DNS by search engines to perform this function has been termed the "Google effect" (see Gillmor, 2003). However, very few provide unbiased results and many question for how long will these be able to maintain that policy. Most search engines are owned by investment ventures and their income results from selling specific "keywords" to advertisers who, in return for their payments see their URLs displayed first when some keyword is present in a user's query. This opens an opportunity to create a specialized search engine that could provide a better service to its target audience and attract those interested in addressing the specificities of that audience.

Global search engines are commercially motivated and rank higher the sites that pay more to be listed on search results pages. Some claim that ranking is not affected by these payments. However, for smaller communities results will always differ from expected. As the profile of web starting sites used by the Portuguese is completely different from that of Americans, the popularity of sites that does not take into account this difference will not reflect the ranking that a Portuguese user would expect. That is probably why when we search for "passaporte" or "biblioteca" ("passport" and "library" in English) we see so unexpected results when going to global web search engines.

In addition to providing information to users about documents that match their queries, search engines have several other functions. Some of these will be important for the Portuguese national interests:

- Advertising of products of services of interest to the Portuguese community
- Conducting sociologic studies based on studying what people look for on the Internet and what sites they are visiting.
- Preserving the language in the digital realm, by providing a tool for searching data on the web written in Portuguese and data to be used by computational linguistics tools.
- Web Archiving. Who is preserving contents of historical relevance being put on-line for use by history researchers 50 years from today?

In addition, most Portuguese web sites, in particular those from the public sector do not offer a search facility on their sites. Tumba! could provide these services to the potentially interested entities, at a much smaller total cost of ownership.

Tumba! has a cache feature that gives acess to last crawled version of each page in the public site (as available on the Google search engine). The internal search interface of tumba! enables internal users to visualize previously crawled versions of each page (our University research group does not have a mandate to publish this information).

6 Conclusion

This paper presented the case for specialised, community search engines. The discussion was centred on the reasons that motivate the creation of a search engine for the Portuguese web. These include, among others, the archival of the information published for preservation, ability for providing independent search services for national public entities, and providing data for sociologic, linguistic and cultural research studies of the Portuguese community. It also detailed the global architecture and main design decisions of tumba!, showing that such community web search engines can be effectively built and operated with moderate resources.

Tumba! is quickly evolving to become an information service that combines database and information retrieval query processing with context and knowledge of data semantics to provide optimised results to Portuguese users.

There is an opportunity for making more specialized information location mediators for the users of thematic, language-specific or cultural webs. These users have special information needs that cannot be met by an engine that, given its global nature, makes no or little use of contextual information while organizing search results. Tumba! was built from the start having in mind that it would be extensible and should provide support for indexing and searching XML data making use of meta-data in RDF (Resource Description Framework) format. The Semantic Web is the new vision for the organization of the next generation of the Web. As this vision unfolds into an extended large-scale database of semantically rich data, search engines will have the foundations for providing more relevant results.

Tumba! does not offer an index to search historical data. The existing index is recreated periodically from the start with the last version of each crawled URL. We have no knowledge of an efficient index structure that could be incrementally updated and

support query restrictions based on the crawling dates of each page. This would be an interesting addition for users willing to navigate on the historical data available on the tumba! repository.

6.1 Acknowledgements

Tumba! is not an individual project, but rather the result of extensive interactions with a group of collaborators. Several members of the XLDB research group have participated in the discussions whose outcome was the design presented in this paper, and developed the system currently in operation. Among these, João Campos, Miguel Costa, Daniel Gomes, Bruno Martins and Norman Noronha developed most of the system now in operation.

The operation of tumba! and its hardware are supported by the FCCN –Fundação para a Computação Científica Nacional[11]. The XMLBase project is supported by Fundação para a Ciência e Tecnologia, under grant POSI / SRI / 40193 / 2001.

7 References

Aires, R. and Aloísio, S. and Quaresma, P. and Santos, D. and Silva, Mário J. (2003). *An initial proposal for cooperative evaluation on information retrieval in Portuguese.* PROPOR'2003 - 6th Workshop on Computational Processing of the Portuguese Language - Written and Spoken, Faro Portugal (accepted for publication).

Arasu, Arvind and Cho, Junghoo and Garcia-Molina, Hector and Paepcke, Andreas and Raghavan, Sriram, (2001). *Searching the Web,* ACM Transactions on Internet Technology.

Baeza-Yates, Ricardo and Ribeiro-Neto, Berthier, (1999). *Modern Information Retrieval*, Addison-Wesley.

Bharat, K. and Henzinger, M.R. (1998). *Improved Algorithms for Topic Distillation in a Hyperlinked Environment*. In Proceedings of SIGIR - Conference on Research and Development in Information Retrieval.

[11] see http://www.fccn.pt

Brin S. and Page L. (1998). *The Anatomy of a Large-Scale Hypertextual Web Search Engine*. Proceedings of the Seventh World Wide Web Conference (WWW7), Brisbane, also in a special issue of the journal Computer Networks and ISDN Systems, Volume 30, issues 1-7.

Campos, João, (2002). *Versus: a Web Repository*, Master Dissertation, Faculdade de Ciências da Universidade de Lisboa. Also available as FCUL/DI technical report, http://www.di.fc.ul.pt/tech-reports.

Costa, Miguel (2003). *SIDRA: Web Indexing and Ranking System*, Master Dissertation, Faculdade de Ciências da Universidade de Lisboa, (in preparation).

Costa, Miguel and Silva, Mário J. (2001). *Ranking no Motor de Busca TUMBA*, CRC'01 - 4ª Conferência de Redes de Computadores, Covilhã, Portugal (in Portuguese).

Day, Michael, (2003). *Collecting and Preserving the World Wide Web*. http://www.jisc.ac.uk/uploaded_documents/archiving_feasibility.pdf

Dean, J., and Henzinger, M. R., (1999). *Finding related pages in the World Wide Web*. In Proceedings of WWW-8, the Eighth International World Wide Web Conference.

Fuchs, R., and Martins, B. and Silva, Mário J. (2003). *Statistics of the Tumba! Corpus*, Technical Report. Universidade de Lisboa, Faculdade de Ciências, Departamento de Informática (in preparation).

Gibson, D., Kleinberg, J., and Raghavan, P. (1998). *Inferring web communities from link topology*. In Proceedings, 9th ACM Conference on Hypertext and Hypermedia,NewYork,NY.ACM, ACM.

Gillmor, David, (2002). *'Google effect' reduces need for many domains.* SilliconValley.com, January, 12, 2002.

Gomes, Daniel and Silva, Mário J., (2001). *Tarântula - Sistema de Recolha de Documentos da Web*, CRC'01 - 4ª Conferência de Redes de Computadores.

Hearst, M. A. and Pedersen J., (1996). O. *Reexamining the Cluster Hypothesis: Scatter/Gather on Retrieval Results*. Proceedings of 19th ACM SIGIR Conference on Research and Development in Information Retrieval.

Kahle, B. (1997). *Preserving the Internet*. Scientific American, 276 (3), 72-73.

Kumar, R. and Raghavan, P. and Rajagopalan, S. and A. Tomkins, (1999). *Trawling the Web for emerging cyber-communities*. In Proceedings of the 8th International World Wide Web Conference.

Henzinger, Monika R. and Motwani, Rajeev and Silverstein, Craig (2002). *Challenges in Web Search Engines*, SIGIR Forum, 36(2).

Katz, Randy, (1990). *Towards a Unified Framework for Version Modeling in Engineering Databases*. ACM Computing Surveys, 22(4):375—408.

Kleinberg, J. (1998), *Authoritative sources in a hyperlinked environment*, in `Proceedings ACM-SIAM Symposium on Discrete Algorithms', San Francisco, California, pp. 668--677

Kobayashi, M. and Takeda, K., (2000). *Information Retrieval on the Web*," ACM Computing Surveys, vol. 32, no. 2, pp. 144--173.

Martins, Bruno, (2003). *Clustering Algorithms for Web Information Retrieval*, Master Dissertation, Faculdade de Ciências da Universidade de Lisboa, (in preparation).

Noronha, Norman, and Campos, João P. And Gomes, Daniel and Silva, Mário J., and Borbinha, José Luís, (2001). *A Deposit for Digital Collections*. Proceedings of the European Conference on Digital Libraries, ECDL, pp. 200-212.

Najork, M., and Heydon, A (2001). *High-performance Web crawling*. Tech. Rep. Research Report 173, Compaq SRC.

Salton, G., (1989). *Automatic Text Processing*. Addison-Wesley.

Silva, M. and Afonso, A. P. (1999). *Designing Information Appliances using a Resource Replication Model*, International Symposium on Handheld and Ubiquitous Computing, HUC'99.